heig-vd
Haute Ecole d'Ingénierie et de Gestion du Canton de Vaud

Introduction à la Logique Mathématique

Profs. Peña & Perez-Uribe & Mosqueron
Basé sur les cours des Prof. F. Sturm (INSA, Lyon) & C. Troestler (Mons, BE)

Polycopié : Electronique numérique

- Portes logiques et algèbre de Boole

  chapitre 4, pages 35 à 54

- Symboles utilisés

  chapitre 4-9, pages 46 et 47

Logique

C'est l'étude des règles formelles que doit respecter toute argumentation correcte

Logique

- La logique a été fondée il y a plus de 2000 ans par Aristote
- Au XVIII siècle, le philosophe E. Kant se sentait autorisé à écrire qu'elle était désormais « close et achevée »
- A la suite des travaux de G. Boole, A. De Morgan et de G. Frege, elle reprit vie au XIX siècle
- Au XX siècle le Théorème d'incomplétude de Gödel (1931) exprime qu'il existe des propositions (en arithmétique) qu'on ne peut ni démontrer, ni réfuter.

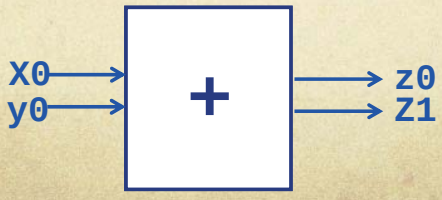
- « Le fil conducteur de Magritte est la transgression de la logique rationaliste parce qu'elle limite notre perception du réel »

Pourquoi étudier la logique mathématique ?

- Comprendre la nature intime du raisonnement
- Donner un sens précis à ce que peut-être le vrai dès qu'il s'agit de raisonnement et d'argumentation
- Pour les informaticiens:
  - Mécaniser le processus de raisonnement
  - Formaliser les objets informatiques (pour la sureté et la sécurité)
  - C'est le langage mathématique de base de tout système informatique

Pourquoi étudier la logique mathématique ? (2)

- La vérification formelle consiste soit à s'assurer que des propriétés spécifiques sont bien respectées par le système construit, soit à s'assurer que deux systèmes sont fonctionnellement équivalents.
- Contrairement à la vérification traditionnelle basée sur l'expérimentation, la vérification formelle est basée sur la démonstration logique ou mathématique.



Logique propositionnelle

- En mathématique, une proposition (ou un résultat mathématique) est un énoncé susceptible d'être démontré ou réfuté.
- Suivant son importance, il est qualifié de:
  - lemme: résultat d'une importance mineure
  - théorème: résultat d'une importance majeure
- Faire une démonstration (on dit aussi une preuve), c'est réaliser un processus qui permet de passer de propositions supposées vraies prises comme hypothèses à une proposition appelée conclusion en utilisant des règles strictes de logique.

Opérateurs logiques

- Les connecteurs logiques ou opérateurs logiques établisent une liaison entre deux énoncés. Ils permettent de créer de nouveaux prédicats (dits prédicats composés) à partir de prédicats P, Q, etc.
- Exemples:
  - Négation: (¬ P)
  - Conjonction (P ∧ Q)
  - Disjonction (P ∨ Q)
  - Implication (P => Q)
  - Equivalence (P ≡ Q)

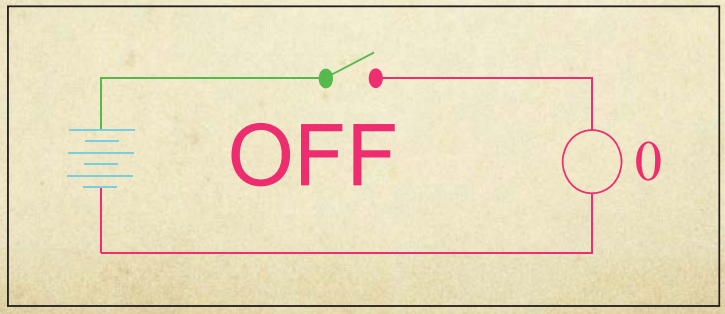
heig-vd
Haute Ecole d'Ingénierie et de Gestion du Canton de Vaud

Algèbre Booléene et Systèmes combinatoires

Profs. Peña & Perez-Uribe & Mosqueron
Basé sur le cours du Prof. E. Sanchez

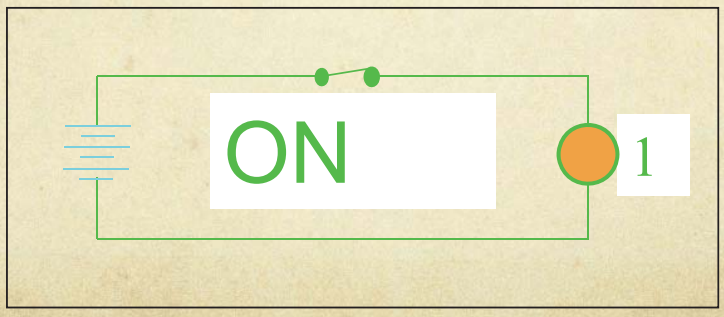
Principe de la logique (postulat)

- Être logique, c'est
  avoir une réponse unique sans contradiction
  - Pas d'Affirmation et de Négation en même temps !!!
    - Une lampe ne peut jamais être Allumée (ON) et Eteinte (OFF) en même temps



Principe de la logique (postulat)

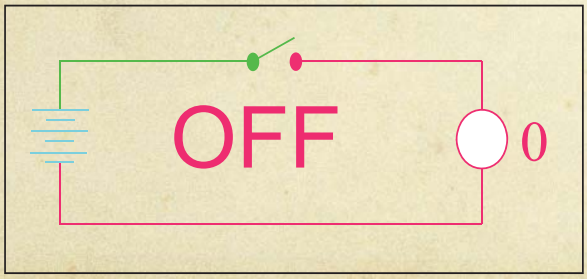
- Être logique, c'est
  avoir une réponse unique sans contradiction
  - Pas d'Affirmation et de Négation en même temps !!!
    - Une lampe ne peut jamais être Allumée (ON) et Eteinte (OFF) en même temps



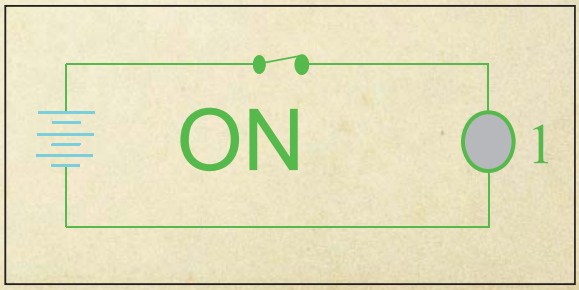
Principe de la logique (postulat)

- On voit clairement une Variable binaire :
  - symbolisé par les états '0' et '1'

0 ➔ OFF



1 ➔ ON



Système logique

- C'est un système qui traite l'information de façon digitale
- Pour étudier un système logique, il faut connaître les éléments de base (les composants) et le langage mathématique qui permet d'écrire les équations de comportement
- Pour un additionneur:

X	Y	Z
0	0	0
0	1	1
1	0	1
1	1	0

$Z = f(X, Y)$

Système binaire

- Système digital qui emploie des signaux à deux valeurs uniques
- En général, les digits employés sont 0 et 1, qu'on appelle bits (binary digits)
- Avantages:
  - on peut utiliser des interrupteurs comme éléments de base du système
  - un signal binaire est plus fiable qu'un autre à plus d'états
  - les décisions prises dans un système digital sont très souvent binaires

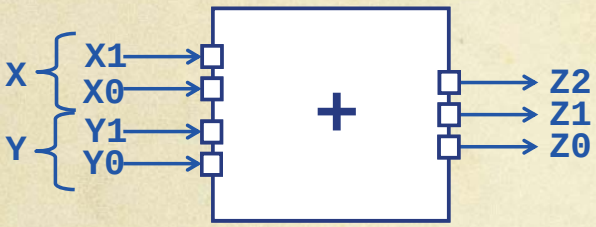
Définitions

- Etat logique :
  - chacune des 2 valeurs que peut prendre une variable logique
- Variable logique :
  - grandeur qui ne peut prendre que les 2 états logiques
- Variable d'entrée (ou simplement entrée) :
  - information à 2 états reçue par un système logique
- Variable de sortie (ou simplement sortie) :
  - information à 2 états générée par un système logique
- Fonction logique :
  - relation logique entre une sortie et une ou plusieurs entrées

Types de systèmes logiques

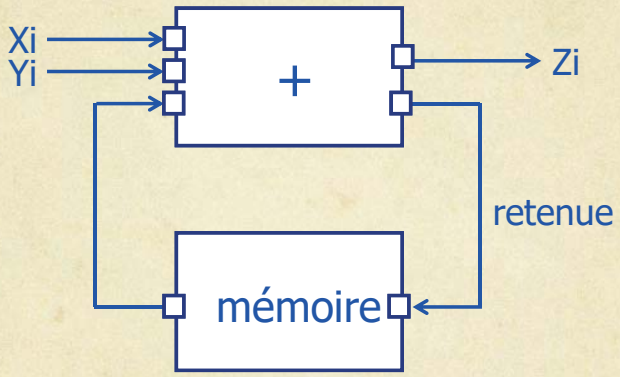
- **Système combinatoire:**
  - la valeur des sorties à un moment donné dépend uniquement des valeurs des entrées à cet instant
  - le comportement est entièrement décrit par une table, la table de vérité, où pour chaque combinaison des entrées on donne la valeur des sorties
  - pour n entrées, la table de vérité comporte 2^n lignes
  - la sortie est immédiate
- **Système séquentiel:**
  - la valeur des sorties dépend de l'histoire des entrées, de leur séquence dans le temps
  - l'obtention d'un résultat peut demander plusieurs étapes
  - le système doit se rappeler des résultats intermédiaires: il faut une mémoire

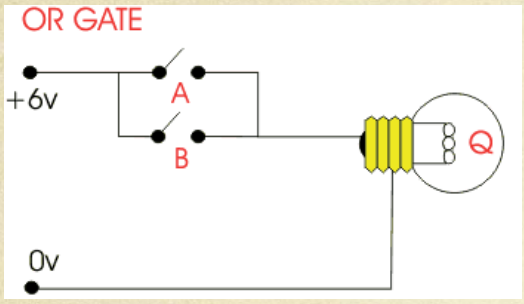
Additionneur combinatoire

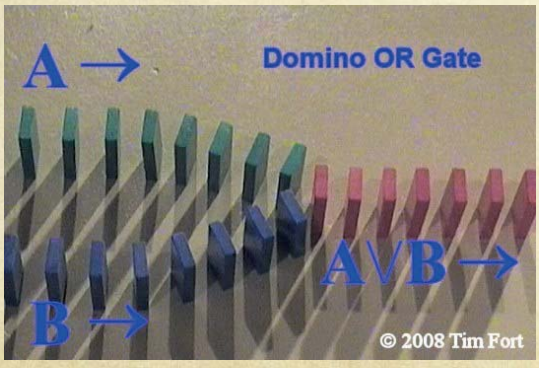


X1	X0	Y1	Y0	Z2	Z1	Z0
0	0	0	0	0	0	0
0	0	0	1	0	0	1
0	0	1	0	0	1	0
0	0	1	1	0	1	1
0	1	0	0	0	0	1
0	1	0	1	0	1	0
0	1	1	0	0	1	1
0	1	1	1	1	0	0
1	0	0	0	0	1	0
1	0	0	1	0	1	1
1	0	1	0	1	0	0
1	0	1	1	1	0	1
1	1	0	0	0	1	1
1	1	0	1	1	0	0
1	1	1	0	1	0	1
1	1	1	1	1	1	0

Additionneur séquentiel







Les portes logiques de bases

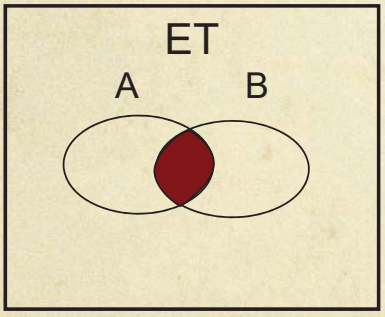
- Etude des fonctions d'une variable
  Nous verrons la porte logique de base :
  - NON
- Etude des fonctions de deux variables
  Nous verrons les portes logiques de base :
  - ET, OU

  nous verrons ensuite des combinaisons :
  - NON-ET, NON-OU
  - OU-Exclusif

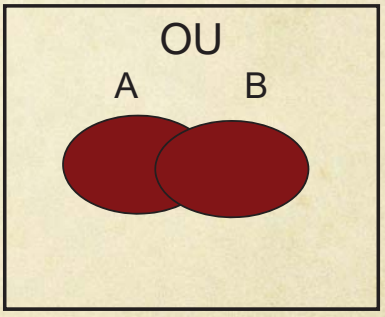
Diagramme de Venn

Représentation graphique d'une fonction logique

ET: intersection



OU: réunion



Fonctions d'une variable

Table des fonctions d'une variable :

Variable	Fonctions F1.x			
A	F1.0	F1.1	F1.2	F1.3
0	0	0	1	1
1	0	1	0	1

F1.0 = 0 constante

F1.1 = A transmission

F1.2 = not A = /A

F1.3 = 1 constante

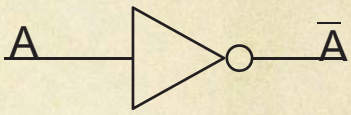
Seule fonction non triviale d'une seule variable :
le **NON** (inversion logique)

Fonction NON (not)

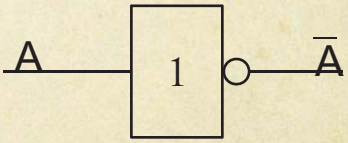
- F1.2 = not A = /A
  - La sortie du circuit est à l'état logique inverse de son entrée

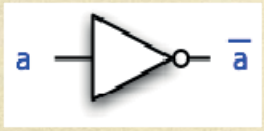
A	F1.2 $\bar{a}$
0	1
1	0

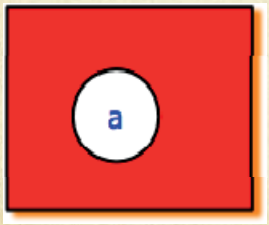
Symbole MIL (US)



Symbole IEEE







Fonctions de deux variables ...

Table des 16 fonctions de deux variables (fcts 0 à 7)

Variables		Fonctions F2.x							
A	B	F2.0	F2.1	F2.2	F2.3	F2.4	F2.5	F2.6	F2.7
0	0	0	0	0	0	0	0	0	0
0	1	0	0	0	0	1	1	1	1
1	0	0	0	1	1	0	0	1	1
1	1	0	1	0	1	0	1	0	1

and xor or

F2.0 = 0

F2.1 = A and B = A • B

F2.2 = A and not B = A • /B

F2.3 = A

F2.4 = not A and B = /A • B

F2.5 = B

F2.6 = A xor B = A ⊕ B

F2.7 = A or B = A + B

Fonctions de deux variables ...

Table des 16 fonctions de deux variables: (fcts 8 à 15)

Variables		Fonctions F2.x							
A	B	F2.8	F2.9	F2.A	F2.B	F2.C	F2.D	F2.E	F2.F
0	0	1	1	1	1	1	1	1	1
0	1	0	0	0	0	1	1	1	1
1	0	0	0	1	1	0	0	1	1
1	1	0	1	0	1	0	1	0	1

nor nand

F2.8 = /F2.7 = /(A + B)

F2.9 = /F2.6 = /(A ⊕ B)

F2.A = /F2.5 = /B

F2.B = /F2.4

F2.C = /F2.3 = /A

F2.D = /F2.2

F2.E = /F2.1 = /(A • B)

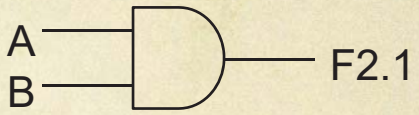
F2.F = /F2.0 = 1

Fonction ET (and)

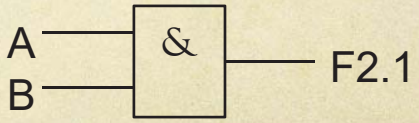
- F2.1 = A . B = A and B
  - La sortie de la porte « ET » est à 1 si l'entrée A et l'entrée B sont à 1

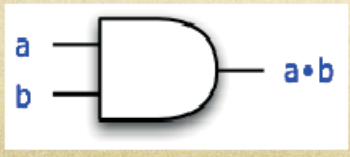
A	B	a•b F2.1
0	0	0
0	1	0
1	0	0
1	1	1

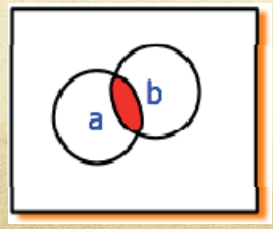
Symbole MIL (US)



Symbole IEEE

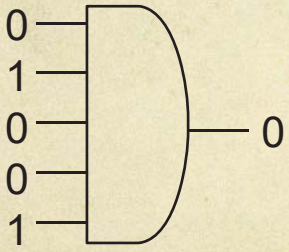


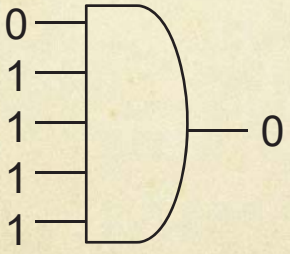


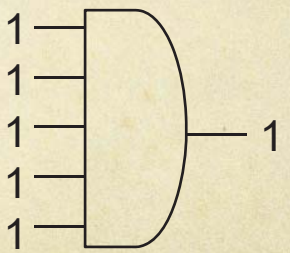


Fonction ET à n entrées

- La définition peut-être étendue à n entrées:
  - La sortie de la porte « ET » est à 1 si toutes les entrées sont à 1.





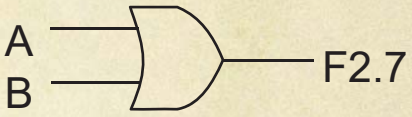


Fonction OU (or)

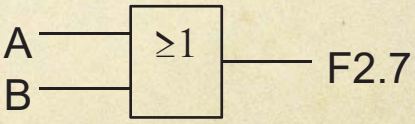
- F2.7 = A + B = A or B
  - La sortie de la porte « OU » est à 1 si l'entrée A ou B valent 1 (l'une ou l'autre ou les deux)

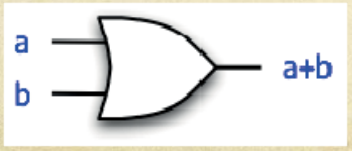
A	B	a+b F2.7
0	0	0
0	1	1
1	0	1
1	1	1

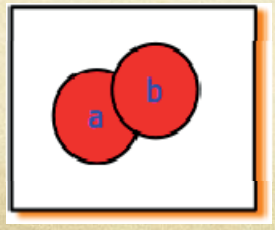
Symbole MIL (US)



Symbole IEEE

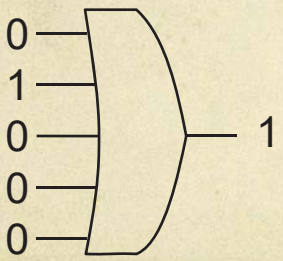


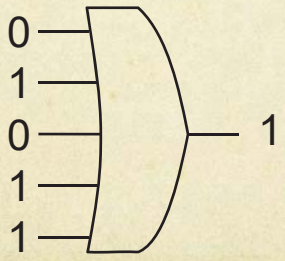


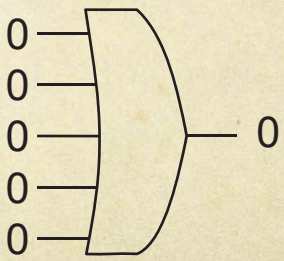


Fonction OU à n entrées

- La définition peut-être étendue à n entrées:
  - La sortie du circuit «OU» est à 1 si une entrée est à 1.



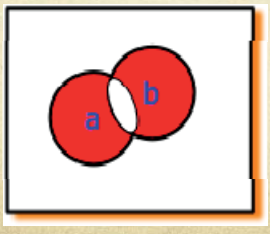




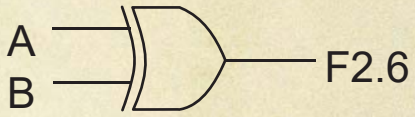
Fonction OU-Exclusif (xor)

- F2.6 = A ⊕ B = A xor B
  - La sortie de la porte « OU-Exclusif » est à 1 si A ou B valent 1, mais pas les 2 (détecte la différence)
  - Pas une fonction de base!

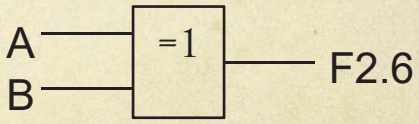
A	B	a⊕b F2.6
0	0	0
0	1	1
1	0	1
1	1	0



Symbole MIL (US)



Symbole IEEE

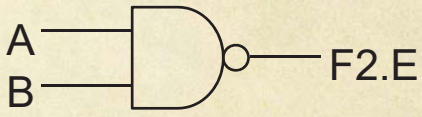


Fonction NON-ET (nand)

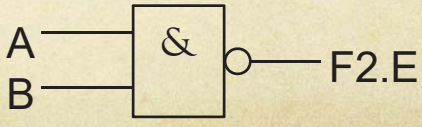
- F2.E = /F2.1 = /(A . B) = A nand B
  - Fonction inverse du ET, soit:
    la fonction est à 1 si une entrée est à 0 .
  - Fonction universelle

A	B	F2.1	F2.E
0	0	0	1
0	1	0	1
1	0	0	1
1	1	1	0

Symbole MIL (US)



Symbole IEEE

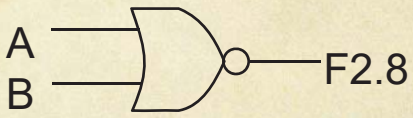


Fonction NON-OU (nor)

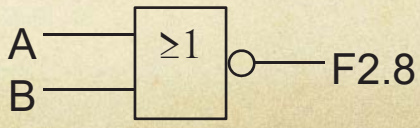
- F2.8 = /F2.7 = /(A + B) = A nor B
  - Fonction inverse du OU, soit:
    la fonction est à 1 si toutes les entrées sont à 0
  - Fonction universelle

A	B	F2.7	F2.8
0	0	0	1
0	1	1	0
1	0	1	0
1	1	1	0

Symbole MIL (US)



Symbole IEEE



Réalisation d'un système combinatoire

- Tout système combinatoire, quelque soit le nombre d'entrées, peut être décrit avec les 3 fonctions de base :
  - inversion
  - ET à 2 entrées
  - OU à 2 entrées
- ou seulement avec la fonction universelle NAND à 2 entrées
- ou seulement avec la fonction universelle NOR à 2 entrées

Logigramme

- Logigramme = schéma logique
- Utilise les symboles graphiques des fonctions usuelles (ET, OU, ...)
- Montre les liaisons entre les entrées, les fonctions utilisées et les sorties
- Par convention, les signaux vont de gauche à droite (entrées à gauche, sorties à droite)

Exercices I

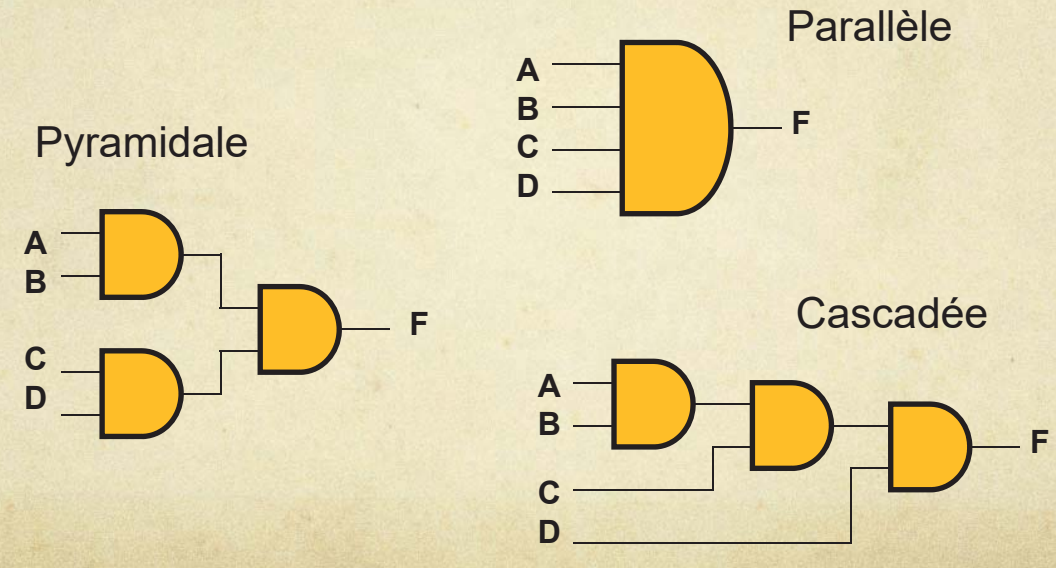
- Ecrivez l'équation logique des sorties Z1 et Z0 de l'additionneur 2 bits de la page 7.
- Exprimez la fonction OU-EXCLUSIF à 2 entrées à l'aide des fonctions de base uniquement, dessinez le logigramme.
- Pourquoi la fonction XNOR est-elle appelée « égalité »?
- Réalisez la fonction « impair » à 3 entrées, en utilisant des fonctions OU-EXCLUSIF à 2 entrées.
- Démontrez que les fonctions NAND et NOR sont universelles.

Décomposition fonctions de base

- Les fonctions ET, OU à plus de 2 entrées peuvent être réalisées à l'aide des fonctions correspondantes à 2 entrées seulement
- 3 formes :
  - Parallèle (non décomposée)
  - Décomposition pyramidale
  - Décomposition en cascade

Décomposition fonctions de base

- Exemple : fonction ET à 4 entrées



Exercice: Additionneur 1 bit

C	X	Y	S	Cout
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

S= C⊕X ⊕Y
Cout= XY+XC+YC

Algèbre de Boole : postulats

- Elle fut initiée en 1854 par le mathématicien britannique George Boole.
- Postulats de l'algèbre de Boole

$$a \bullet \overline{a} = 0$$

$$a + \overline{a} = 1$$

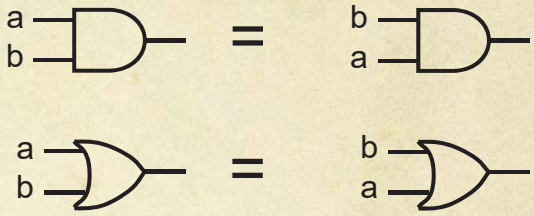
découlent de l'hypothèse :

l'inverse d'une variable ne peut jamais avoir la même valeur que la variable

Algèbre de Boole

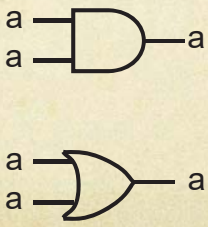
- Commutativité:

$$a \bullet b = b \bullet a$$
$$a + b = b + a$$



- Idempotence:

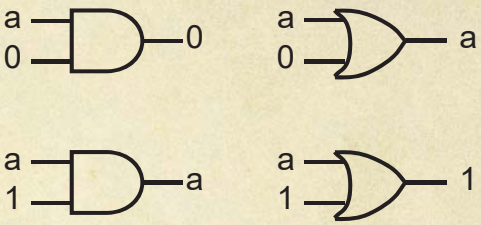
$$a \bullet a = a$$
$$a + a = a$$



Algèbre de Boole

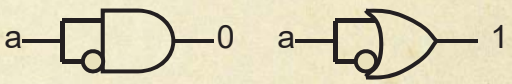
- Constantes:

$$a \bullet 0 = 0 \qquad a + 0 = a$$
$$a \bullet 1 = a \qquad a + 1 = 1$$



- Complémentation:

$$a \bullet \bar{a} = 0$$
$$a + \bar{a} = 1$$

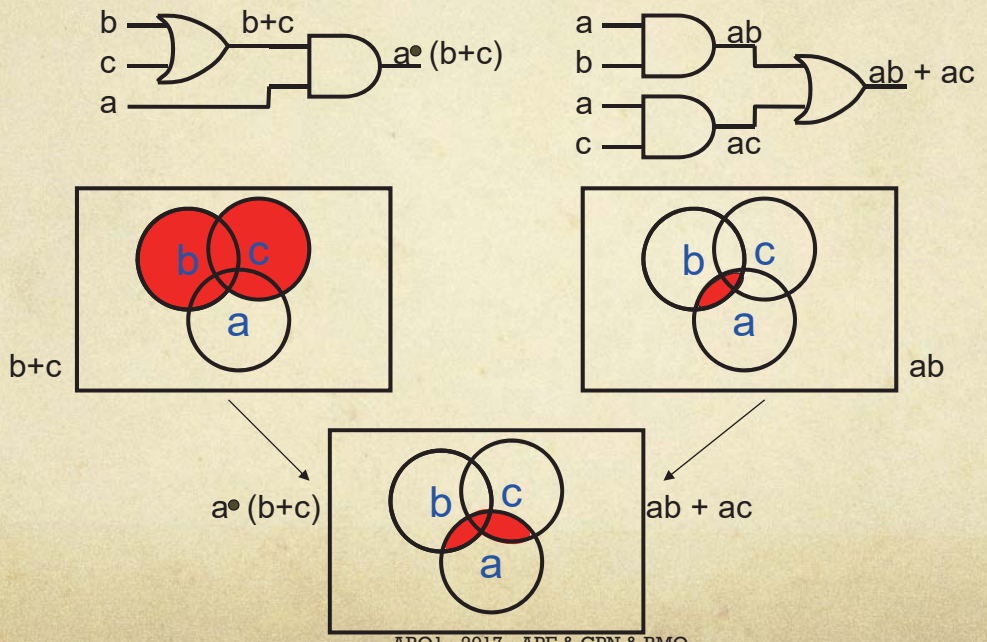


Algèbre de Boole

- Distributivité:

$$a \bullet (b + c) = (a \bullet b) + (a \bullet c)$$

$$a + (b \bullet c) = (a + b) \bullet (a + c)$$

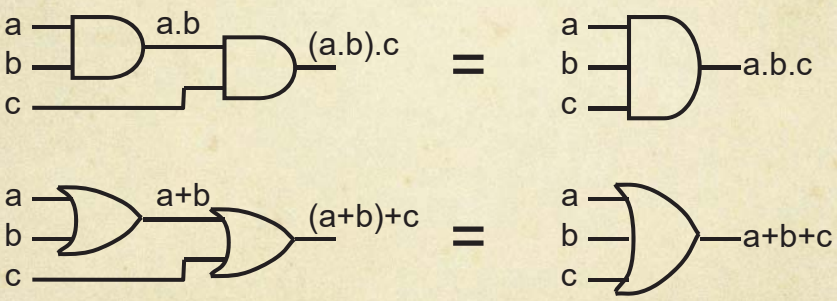


Algèbre de Boole

- Associativité:

$$a \bullet (b \bullet c) = (a \bullet b) \bullet c = a \bullet b \bullet c$$

$$a + (b + c) = (a + b) + c = a + b + c$$

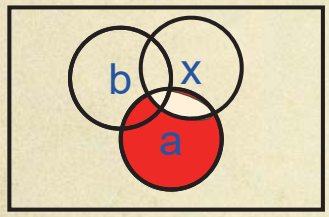


Algèbre de Boole

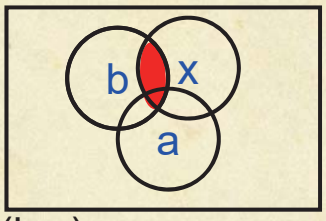
- Consensus:

$$(a \bullet \bar{x}) + (b \bullet x) + (a \bullet b) = (a \bullet \bar{x}) + (b \bullet x)$$

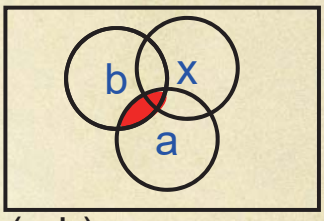
$$(a + \bar{x}) \bullet (b + x) \bullet (a + b) = (a + \bar{x}) \bullet (b + x)$$



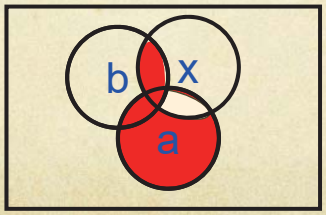
$(a \bullet \bar{x})$



$(b \bullet x)$



$(a \bullet b)$

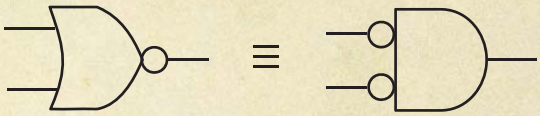


$(a \bullet \bar{x}) + (b \bullet x) + (a \bullet b)$

Théorèmes de De Morgan (1806-1871)

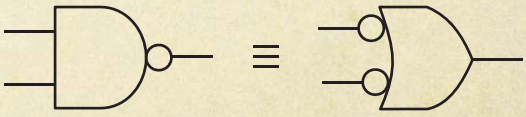
XIV /(A + B) = /A • /B

A	B	/A	/B	F2.8
0	0	1	1	1
0	1	1	0	0
1	0	0	1	0
1	1	0	0	0



XV /(A • B) = /A + /B

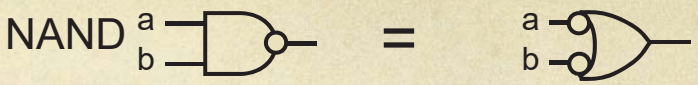
A	B	/A	/B	F2.E
0	0	1	1	1
0	1	1	0	1
1	0	0	1	1
1	1	0	0	0

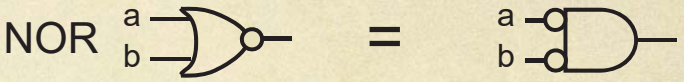


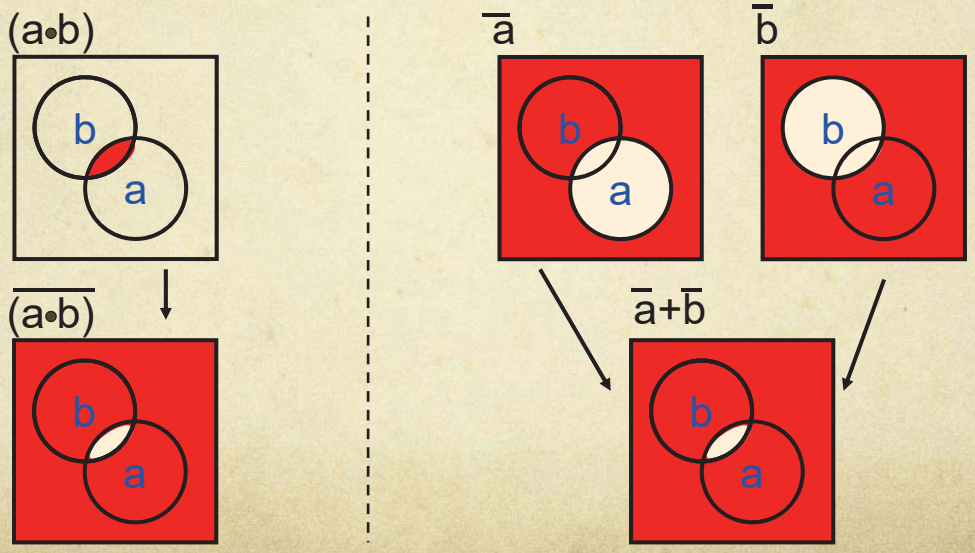
Théorèmes de De Morgan (1806-1871)

$$\overline{(a \bullet b)} = \bar{a} + \bar{b}$$

$$\overline{(a + b)} = \bar{a} \bullet \bar{b}$$



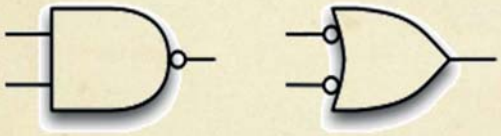




Fonctions complètes

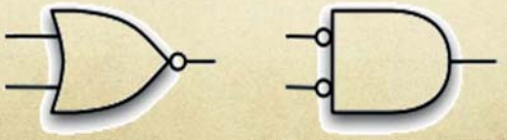
- Un opérateur est complet lorsqu'il permet la réalisation des trois fonctions logiques de base (NON, ET, OU)
- NAND

$$a \uparrow b = \overline{(a \bullet b)} = \bar{a} + \bar{b}$$



- NOR

$$a \downarrow b = \overline{(a + b)} = \bar{a} \bullet \bar{b}$$



Formules

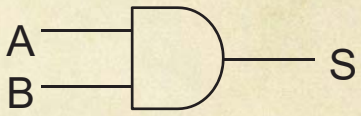
- Démontrer:
  - A + AB = A
  - A + /AB = A + B
  - (A+B).(A+C) = A + BC

Théorème	Forme Somme logique	Forme Produit logique
Elément absorbant	$A + 1 = 1$	$A \cdot 0 = 0$
Elément neutre	$A + 0 = A$	$A \cdot 1 = A$
Complémentation	$A + \overline{A} = 1$	$A \cdot \overline{A} = 0$
Idempotence	$A + A = A$	$A \cdot A = A$
Associativité	$(A+B)+C = A+(B+C) = A+B+C$	$(A.B).C = A.(B.C) = A.B.C$
Commutativité	$A+B = B+A$	$A.B = B.A$
Distributivité	$A.(B+C) = A.B + A.C$	$A+(B.C) = (A+B).(A+C)$
Absorption	$A+A.B = A$	$A.(A+B) = A$
Simplification	$A + \overline{A}.B = A + B$	$A.(\overline{A} + B) = A.B$
Involution	$\overline{\overline{A}} = A$	
Lois de De Morgan	$\overline{A + B} = \overline{A} . \overline{B}$	$\overline{A.B} = \overline{A} + \overline{B}$

Table de vérité (TDV)

- Liste des valeurs de sortie en fonction des combinaisons des entrées

no	B	A	S
0	0	0	0
1	0	1	0
2	1	0	0
3	1	1	1



- Permet de spécifier touts les états d'une fonction logique => cahier des charges

Mintermes

- Un minterme de n variables est un monôme possédant les n variables, sous forme vraie ou inversée.
  Il existe un minterme par état d'entrée d'une fonction combinatoire (ou ligne de la table de vérité)
- Exemple pour n=4

  minterme 0 = /B • /A

  minterme 1 = /B • A

  minterme 2 = B • /A

  minterme 3 = B • A

Construction TDV

- Table avec la liste de toutes les combinai-sons des entrées
  - N entrées => 2^N lignes dans la table

No minterme	D	C	B	A	F
0	0	0	0	0	1
1	0	0	0	1	0
2	0	0	1	0	0
..	..	..	..	..	..
14	1	1	1	0	1
15	1	1	1	1	0

Liste des mintermes d'une fonction

- Soit la TDV d'une fonction :

No	C	B	A	F
0	0	0	0	1
1	0	0	1	0
2	0	1	0	0
3	0	1	1	1
4	1	0	0	0
5	1	0	1	1
6	1	1	0	0
7	1	1	1	1

Nous pouvons résumer la TDV en donnant la liste des mintermes vrai :

$$F(C,B,A) = \Sigma\ 0, 3, 5, 7$$

Forme canonique décimale

Forme canonique algébrique

- Toute fonction logique combinatoire peut être exprimée comme une somme de mintermes, ceux où la fonction est égale à 1: c'est la forme canonique algébrique, unique pour une fonction donnée
- Un monôme est un produit logique de n variables, vraies ou inversées
- Un polynôme est une somme logique de plusieurs monômes

Equation logique

- L'équation canonique découle directement de la TDV. Mais il peut exister des solutions équivalentes.
- Exemple: la fonction OU

B	A	Z
0	0	0
0	1	1
1	0	1
1	1	1

équation canonique:

$Z(B,A) = \overline{B}A + B\overline{A} + BA$

équation simplifiée:

$Z(B,A) = B + A$

- Comment simplifié la fonction => deux méthodes:
  - algébrique (algèbre de Boole)
  - graphique (table de Karnaugh)

Forme canonique décimale

- Si chaque minterme est remplacé par la valeur décimale correspondante à la combinaison binaire de ses variables (1 si la variable est vraie et 0 si elle est inversée), on obtient la forme canonique décimale d'une fonction logique combinatoire.
  Dans ce cas, il est impératif de préciser l'ordre et le nombre des variables.

Forme canonique décimale

- Liste des mintermes d'une fonction
- Soit la TDV d'une fonction :

No	C	B	A	F
0	0	0	0	1
1	0	0	1	0
2	0	1	0	0
3	0	1	1	1
4	1	0	0	0
5	1	0	1	1
6	1	1	0	0
7	1	1	1	1

Nous pouvons résumer la TDV en donnant la liste des mintermes vrai :

$F(C,B,A) = \Sigma\ 0, 3, 5, 7$

Forme canonique décimale

Exemple

- Fonction majorité:
  la sortie vaut 1 si une majorité des entrées possède la valeur 1
- Table de vérité pour la majorité de 3 variables:

a b c	MAJ(a,b,c)
0 0 0	0
0 0 1	0
0 1 0	0
0 1 1	1
1 0 0	0
1 0 1	1
1 1 0	1
1 1 1	1

Représentations

- Forme canonique algébrique:

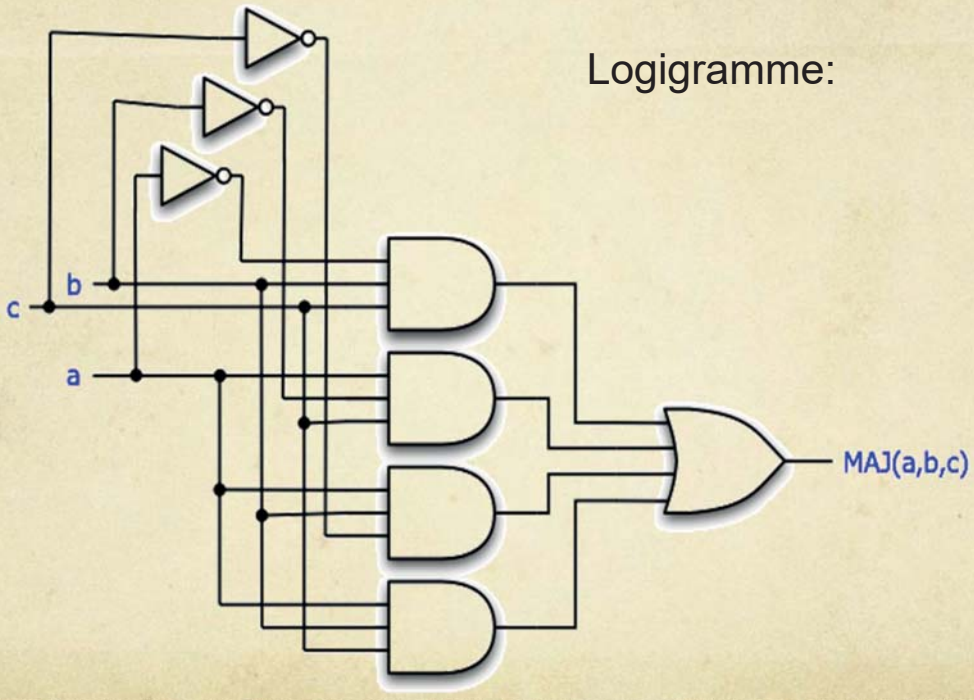
$$MAJ(a,b,c) = \bar{a}bc + a\bar{b}c + ab\bar{c} + abc$$

- Forme canonique décimale:

$$MAJ(a,b,c) = \sum 3,5,6,7$$

Représentations

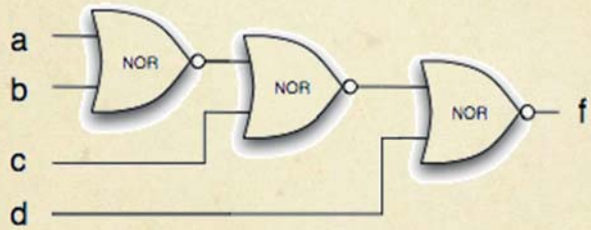
Logigramme:



Analyse d'un système combinatoire

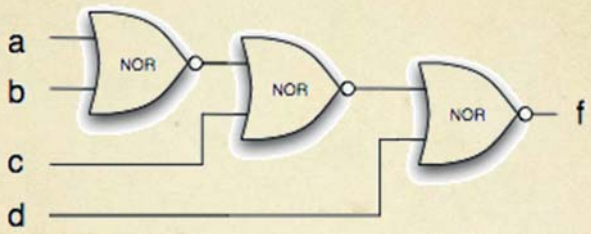
- Analyse: déterminer le comportement d'un système à partir d'une description de sa structure

Exemple:



Donnez l'expression algébrique minimale de f(a,b,c,d), l'expression canonique algébrique, l'expression canonique décimale et la table de vérité.

Expression algébrique minimale



$$
\begin{aligned}
f(a,b,c,d) &= \overline{\overline{\overline{\overline{a+b}+c}+d}} \\
&= (\overline{(a+b)}+c)\bar{d} \\
&= (\bar{a}\bar{b}+c)\bar{d} \\
&= \bar{a}\bar{b}\bar{d}+c\bar{d}
\end{aligned}
$$

Expression canonique algébrique

$$
\begin{aligned}
f(a,b,c,d) &= \bar{a}\bar{b}\bar{d}+c\bar{d} \\
&= \bar{a}\bar{b}\bar{d}(c+\bar{c})+c\bar{d}(a+\bar{a})(b+\bar{b}) \\
&= \bar{a}\bar{b}\bar{d}c+\bar{a}\bar{b}\bar{d}\bar{c}+\left[c\bar{d}a+c\bar{d}\bar{a}\right](b+\bar{b}) \\
&= \bar{a}\bar{b}c\bar{d}+\bar{a}\bar{b}\bar{c}\bar{d}+c\bar{d}ab+c\bar{d}a\bar{b}+c\bar{d}\bar{a}b+c\bar{d}\bar{a}\bar{b} \\
&= \bar{a}bc\bar{d}+\bar{a}\bar{b}\bar{c}\bar{d}+abc\bar{d}+a\bar{b}c\bar{d}+\bar{a}bc\bar{d}+\bar{a}\bar{b}c\bar{d} \\
&= \bar{a}bc\bar{d}+\bar{a}\bar{b}\bar{c}\bar{d}+abc\bar{d}+a\bar{b}c\bar{d}+\bar{a}bc\bar{d}
\end{aligned}
$$

Expression canonique décimale

$$f(a,b,c,d) = \bar{a}\bar{b}c\bar{d} + \bar{a}\bar{b}\bar{c}\bar{d} + abc\bar{d} + a\bar{b}c\bar{d} + \bar{a}bc\bar{d}$$
$$= \sum(2,0,14,10,6)$$
$$= \sum(0,2,6,10,14)$$

Table de vérité

a b c d	f
0 0 0 0	1
0 0 0 1	0
0 0 1 0	1
0 0 1 1	0
0 1 0 0	0
0 1 0 1	0
0 1 1 0	1
0 1 1 1	0
1 0 0 0	0
1 0 0 1	0
1 0 1 0	1
1 0 1 1	0
1 1 0 0	0
1 1 0 1	0
1 1 1 0	1
1 1 1 1	0